

The folarrow package

Arrows with three strokes for the projection of a foliated bundle

Yuwsuke Kieda

2026-10-10 v20261010.0

A Japanese version of this manual is `folarrow-ja.pdf`.

1 Overview

When a fiber bundle is drawn as a diagram, its projection is drawn as a vertical arrow.

$$\begin{array}{ccc} V & \longrightarrow & E \\ & & \downarrow \pi \\ & & M \end{array}$$

For a foliated bundle, three slanted strokes are sometimes drawn across the shaft of this arrow.

$$\begin{array}{ccc} V & \longrightarrow & E \\ & & \downarrow \text{\texttt{\textbackslash}three} \pi \\ & & M \end{array}$$

The `folarrow` package makes such arrows available in `amscd`, `xy-pic` and `tikz-cd` diagrams and in inline math.

The arrows themselves are the usual arrows of each package, and the three strokes are drawn over the middle of the shaft. The strokes are drawn with `pict2e` (PGF in `tikz-cd`), so no additional fonts are needed for them. The length and the line thickness of the strokes follow the size of the current math font (font size and math style).

2 Loading

```
\usepackage{amscd}      % only the ones you use
\usepackage[all]{xy}
\usepackage{tikz-cd}
\usepackage{folarrow}
```

`amscd`, `xy-pic` and `tikz-cd` may be loaded before or after `folarrow` (the support for them is set up at `\begin{document}`).

2.1 Light version: the `lite` option

```
\usepackage[lite]{folarrow}
```

With `lite`, the package uses no other files (the Lua module, the measured data and the arrow fonts) and draws all arrows (there is also a separate manual for the light version, `folarrow-lite-en.pdf`). Use it, for example, when you do not want the arrow fonts embedded in the PDF. Without it (the default), nothing changes.

- No file other than `folarrow.sty` is used, even if installed: not the Lua module `folarrow.lua`, the measured data `folarrow-fonts.def`, the tables of the arrow fonts `folarrow-glyphs.def` and `folarrow-otf.def`, or the arrow fonts and the map file.
- All arrows, including the inline arrows, are drawn (the fonts of section 6.1 are not used, and `glyphs` is always false). The result is an arrow with three strokes drawn over it, so copying it from the PDF gives the original arrow character (for `\foldownarrow`, the pieces of the delimiter).
- The width of the arrowhead and the line thickness (section 8) are read from the glyph bounding box with Xe $\LaTeX$ and an OpenType math font (using `\XeTeXglyphbounds`), and estimated from the font dimensions otherwise (pdf $\LaTeX$, up $\LaTeX$, Lua $\LaTeX$).
- Since the measured data is not used, the strokes are placed at the center of the arrow's box (without correcting for the offset from the center of the ink), and their ends are always butt. For the inline arrows of 10pt Computer Modern, newtxmath, newpxmath and txfonts, the strokes were off the center by up to 0.3pt along the arrow and 0.03pt across it (at most 0.015pt by default). With some fonts the offset is larger, and the width and thickness may also differ from the measured values.
- The workaround of rotating the vertical arrows of Pennstander Math (section 6.1) is not applied.
- `lite` can only be given when loading the package (not with `\folarrowsetup`).

3 amscd

Add `F` right after the `@` of the usual arrows `@>>>`, `@<<<`, `@VVV` and `@AAA` to get the strokes. Labels are given in the same way as for the usual arrows.

```
\begin{CD}
V @>>> E @>f>> E'\\
@VVV @V{\pi}VV @V{\pi'}V\\
M @= M @<g<< M
\end{CD}
```

$$\begin{array}{ccccc} V & \longrightarrow & E & \xrightarrow{f} & E' \\ \downarrow & & \pi \downarrow & & \uparrow \pi' \\ M & \xlongequal{\quad} & M & \xleftarrow[g]{} & M \end{array}$$

4 xy-pic

The package provides a directional object `fol` to be placed on an arrow. Place it as a label on the arrow with `|`, as in `\ar[d]|\@{fol}`. It is an object of size zero, so the arrow is not broken. The direction of the strokes follows the direction of the arrow, and it also works on diagonal arrows.

```
\xymatrix{
V \ar[r] & E \ar[d]|\@{fol}^{\folpad{\pi}} \\
& M }
\quad
\begin{array}{ccc}
E & \xrightarrow{\quad} & B \\
\uparrow & \swarrow & \downarrow \\
M & \xleftarrow{\quad} & N
\end{array}

```

$$\begin{array}{ccc} V & \longrightarrow & E \\ & & \downarrow \pi \\ & & M \end{array} \quad \begin{array}{ccc} E & \xrightarrow{\quad} & B \\ \uparrow & \swarrow & \downarrow \\ M & \xleftarrow{\quad} & N \end{array}$$

A label of the arrow is placed at the position of the strokes, so it may overlap them. Enclose the label in `\folpad`: it adds a margin of half the length of the strokes on all sides, and the label clears the strokes.

```
\ar[d]|\@{fol}^{\folpad{\pi}}
```

5 tikz-cd

Add `foliated` to the options of the arrow.

```
\begin{tikzcd}
V \arrow[r] & E \arrow[d, foliated, "\pi"] \\
& M
\end{tikzcd}
```

$$\begin{array}{ccc} V & \longrightarrow & E \\ & \Downarrow \pi & \\ & M & \end{array} \quad \begin{array}{ccc} E & \dashrightarrow & B \\ \Uparrow & \swarrow & \Downarrow \\ M & \dashleftarrow & N \end{array}$$

`foliated` uses the `decorations.markings` library (loaded by `folarrow`). On arrows with `foliated`, the margin of the labels (`inner sep`) is increased by half the length of the strokes, so that the labels clear the strokes.

6 Inline math

<code>\folrightarrow</code>	$E \dashrightarrow M$
<code>\folleftarrow</code>	$M \dashleftarrow E$
<code>\folddownarrow</code>	$E \Downarrow M$
<code>\foluparrow</code>	$M \Uparrow E$
<code>\xfolrightarrow[below]{above}</code>	$E \dashrightarrow M, E \xrightarrow[\text{proj}]{\pi} M$
<code>\xfolleftarrow[below]{above}</code>	$M \dashleftarrow E$

- All of them are relations (`\mathrel`) and can be used in subscripts: $A_{E \dashrightarrow M}$.
- Base arrows: `\longrightarrow` for `\folrightarrow`, and `\longleftarrow` for `\folleftarrow`.
- `\xfolrightarrow` and `\xfolleftarrow` stretch with the labels above and below, like `\xrightarrow` of `amsmath`, but are never shorter than `\longrightarrow`.

6.1 Fonts for the inline arrows

With the following math fonts, the four arrows `\folrightarrow`, `\folleftarrow`, `\folddownarrow` and `\foluparrow` are typeset with Type 1 fonts in which each arrow, including the strokes, is a single glyph.

Math font	Symbol font	Arrow font
Computer Modern (default)	<code>cmsy5-cmsy10</code>	<code>folcm5-folcm10</code>
Latin Modern (<code>lmodern</code>)	<code>lmsy5-lmsy10</code>	<code>follm5-follm10</code>
<code>newtxmath</code>	<code>ntxsy, ntxsy7, ntxsy5</code>	<code>folntx, folntx7, folntx5</code>
<code>txfonts</code>	<code>txsy</code>	<code>foltx</code>
<code>newpxmath</code>	<code>zplsy</code>	<code>folzpl</code>
<code>pxfonts</code>	<code>pxsy</code>	<code>folpx</code>
Computer Modern (bold)	<code>cmbsy5-cmbsy10</code>	<code>folcmb5-folcmb10</code>
Latin Modern (bold)	<code>lmb5y5, lmb5y7, lmb5y10</code>	<code>folymb5, folymb7, folymb10</code>
<code>newtxmath</code> (bold)	<code>ntxbsy, ntxbsy7, ntxbsy5</code>	<code>folntxb, folntxb7, folntxb5</code>
<code>txfonts</code> (bold)	<code>txbsy</code>	<code>foltxb</code>
<code>newpxmath</code> (bold)	<code>zplbsy</code>	<code>folzplb</code>
<code>pxfonts</code> (bold)	<code>pxbsy</code>	<code>folpxb</code>

- A glyph is the outline of the minus sign and $\rightarrow$ of the symbol font (combined in the same way as `\longrightarrow`) with the outlines of the three strokes added. $\leftarrow$ is made from $\leftarrow$ and the minus sign, and $\downarrow$ and $\uparrow$ are the glyph of $\rightarrow$ rotated, centered on the math axis. The length, spacing, slant, thickness and ends of the strokes are computed from the outline of the original glyph by the same rules as the default when drawing. The center of the strokes is at the center of the arrow's ink.

- The result is a single character of a font, so copying it from the PDF gives U+2192, U+2190, U+2193 and U+2191.
- The font is loaded at the same size as the symbol font, so in subscripts or with `\large` the glyph corresponds to the symbol font of that size.
- When a shape key (section 7) is not at its default, with `glyphs=false`, or with a math font in neither the table above nor the list of OpenType math fonts below, the strokes are drawn over the arrow as before. Then the base of $\downarrow$ and $\uparrow$ is a delimiter 1.25 times the quad of the symbol font high.
- The stretchable arrows (`\xfolrightarrow` etc.) and the arrows of diagrams change their length, so they are always drawn.

OpenType math fonts With Lua $\text{\LaTeX}$ and Xe $\text{\LaTeX}$ and `unicode-math`, the arrows are typeset with OTF fonts (`folu-code.otf`) for the following 49 fonts.

Latin Modern Math, New Computer Modern Math (Regular, Book, Bold), New Computer Modern Sans Math, STIX Two Math, STIX Math, XITS Math (Regular, Bold), $\text{\TeX}$ Gyre Termes, Pagella, Bonum, Schola and DejaVu Math, Libertinus Math, Fira Math, Asana Math, Erewhon Math (Regular, Bold), KpMath (Regular, Light, Semibold, Bold, Sans, SansBold), GFS Neohellenic Math, IBM Plex Math, Arsenal Math (Sans, SansBold), Concrete Math (Regular, Bold), Euler Math, Garamond Math, Lete Sans Math (Regular, Bold), Luciole Math (Regular, Bold), Old Standard Math, XCharter Math (Regular, Bold), Pennstander Math (nine weights from Thin to Black)

- XITS Math Bold has no U+27F6 and U+27F5, so, like `\relbar\joinrel\rightarrow` in $\text{\LaTeX}$, an outline of the minus sign and $\rightarrow$ ($\leftarrow$ and the minus sign) overlapped by 3/18em is used as the long arrow. When drawing, too, arrows are made with `\relbar\joinrel\rightarrow` etc. if the font has no U+27F6 or U+27F5.
- A glyph is the outline of U+27F6 (`\longrightarrow`) or U+27F5 (`\longleftarrow`) of the original font with the outlines of the strokes added. $\downarrow$ and $\uparrow$ are U+27F6 rotated, centered on AxisHeight of the MATH table. The thickness and height of the shaft are measured by cutting the outline of the arrow vertically, and the shape of the tail is also determined from the outline (round if the distance from the end to where the shaft reaches its full width is at least 0.35 times the thickness).
- When the original font has script glyphs (`ssty`) for the arrows (Erewhon Math, Concrete Math, Lete Sans Math, Luciole Math, XCharter Math (Regular only for each), Euler Math, Pennstander Math), glyphs are made from each of them, the font gets the same `ssty` feature, and it is loaded with the same options as the original font (`+ssty=0` etc.).
- The original font is identified by whether the font name part of `\fontname` (a file name or a font name) matches the file name, PostScript name, full name or family name of an original font.
- The characters are at their Unicode positions (U+2192, U+2190, U+2193, U+2191), so as text they are $\rightarrow\leftarrow\downarrow\uparrow$.
- In Pennstander Math, the axis of the shaft piece and that of the head piece of the extensible $\downarrow$ (GlyphAssembly of the MATH table) are misaligned, and a large delimiter $\downarrow$ is broken ($\uparrow$ has no extensible data; this happens without `folarrow` too). So, for the nine fonts of this family, the base of the drawn vertical arrows (`\foldownarrow` and `\foluparrow` with `glyphs=false` etc., and `@FVVV` and `@FAAA` of `amscd`) is not a delimiter but a horizontal arrow (`\longrightarrow` inline, a stretchable arrow as long as `\Big` for `amscd`) rotated by 90°. The center of the ink is taken from the bounding box of the glyph before rotation and is aligned to the math axis. The affected fonts are determined from their extensible pieces and listed in `folarrow-otf.def`. The broken vertical arrows where `folarrow` commands are not used (usual `@VVV` and `@AAA` of `amscd`, `\Big\downarrow`, etc.) are a problem of the font and are not supported.
- U+27F6 and U+27F5 of XCharter Math Bold have ink longer than their advance width (1000), up to 1524, and overlap the next character (a problem of the font). In the OTF, a glyph whose ink exceeds its advance width gets the advance width of the right edge of the ink plus the left side bearing (1580, as in Regular). When drawing, the original width is kept, so they overlap.

- The OTF fonts were made from the original fonts. They keep the licenses of the original fonts, the SIL Open Font License 1.1 or the GUST Font License. Their file names and font names (**FolArrowU-code**) do not contain the names (Reserved Font Names) of the original fonts. There are none for fonts that may not be redistributed, such as Minion Math (the arrows are drawn as before).

The map file of the Type 1 fonts, **folarrow.map**, is loaded by the package with pdf $\LaTeX$ and Lua $\LaTeX$; with dvipdfmx and xdvipdfmx the package issues `\special{pdf:mapfile folarrow.map}` on the first page. For dvips, enable **folarrow.map** with **updmap**.

The Type 1 arrow fonts were made from the original symbol fonts and keep their licenses: SIL Open Font License 1.1 for those from Computer Modern (named **folcm** and **FolArrow-CM** to avoid the Reserved Font Names), GUST Font License for those from Latin Modern, and GPL (with the font exception) for those from txfonts, pxfonts, newtx and newpx.

6.2 Bold

When math is made bold with `\boldmath` (`\mathversion{bold}`) or `\bm` of **bm**, the arrows are typeset with the bold symbol fonts, and the strokes follow them.

- Type 1 math fonts (Computer Modern, Latin Modern, newtx, txfonts, newpx, pxfonts): the inline arrows are typeset with the arrow fonts made from the bold symbol fonts (the bold rows of the table above), and the vertical arrows are bold too. The length and thickness of the strokes of the stretchable arrows and the arrows of diagrams are taken from the measured data of the bold symbol fonts.
- The vertical arrows of **amscd** are large delimiters, so they are bold if the font of large delimiters has a bold version (newtx, txfonts, newpx, pxfonts). **cmex10** and **lmex10** of Computer Modern and Latin Modern have no bold version, so the vertical arrows do not get thicker in bold (as with the usual `\Big\downarrow` of $\LaTeX$). The strokes follow that arrow.
- **unicode-math**: specify the bold math font with **version=bold**, as in

```
\setmathfont{NewCMMath-Bold.otf}[version=bold]
```

and `\boldmath` switches to it. The inline arrows are typeset with the OTF font for the bold font (Bold etc. in the list of section 6.1), and the strokes of the drawn arrows are computed from the glyphs of the bold font. With fonts that have no extensible data for $\downarrow$, such as KpMath Bold and Erewhon Math Bold, the vertical arrows of **amscd** remain short glyphs (like the usual `@VVV`).

- In New Computer Modern Math Bold, the dimensions of the extensible pieces of $\downarrow$ are broken, and Xe $\LaTeX$ does not stop when it tries to build a large delimiter $\downarrow$ (`\Big\downarrow` etc.; this happens without **folarrow** too). So, only with Xe $\LaTeX$, the drawn vertical arrows (`@FVVV` and `@FAAA` of **amscd** etc.) are built from a rotated horizontal arrow, as for Pennstander Math.
- The light version (**lite**) does not use the measured data, so the length and thickness of the strokes in bold are estimates.

Measuring the bold versions of Computer Modern, Latin Modern, newtx, txfonts, newpx and pxfonts typeset with `\boldmath`, the ratio of the computed to the measured width of the arrowhead and thickness of the shaft was 0.99–1.01 for all sizes and styles with pdf $\LaTeX$ and up $\LaTeX$. The offset of the strokes from the center was at most 0.07 pt, except for the stretchable arrows (up to 0.17 pt for bold pxfonts; the limit of `\cleaders` in section 8).

7 Adjusting the strokes

Use package options or `\folarrowsetup`. `\folarrowsetup` inside a group affects only that group.

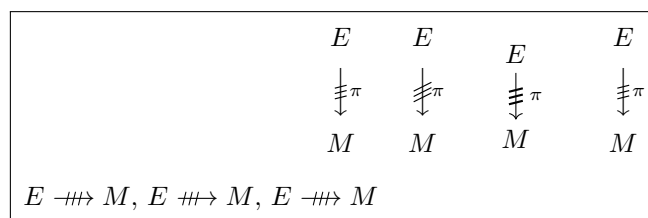
```
\usepackage[gap=0.45]{folarrow}
\folarrowsetup{span=1.2, thickness=0.5pt}
```

Key	Default	Meaning
<code>span</code>	1.0	length of the strokes (ratio to the arrowhead width; $> 0, \leq 10$)
<code>lean</code>	0.25	slant (offset along the arrow relative to the length; -5 to 5)
<code>gap</code>	0.35	spacing of the strokes (ratio to the arrowhead width; 0 to 10)
<code>shift</code>	0	shift from the middle toward the tail (ratio to the arrowhead width; -10 to 10)
<code>thickness</code>	(empty)	line thickness (a length ≥ 0); if empty, the shaft thickness (section 8)
<code>glyphs</code>	true	use the arrow fonts for the inline arrows (section 6.1)
<code>lite</code>	false	light version (only when loading; section 2.1)

Example The following document uses `\folarrowsetup`. It can be copied as it is, from `\documentclass` to `\end{document}`, and compiled (`full-examples/ex-setup-en.tex`; the result is that example actually typeset).

```
\documentclass{article}
\usepackage{amscd}
\usepackage{tikz-cd}
\usepackage{folarrow}
\folarrowsetup{gap=0.45} % whole document: slightly wider spacing of the strokes
\begin{document}
\[
\begin{CD} E @> @FVV{\pi}V @> M \end{CD}
\qqquad
{\folarrowsetup{span=1.5, lean=0.5} % only inside this group
\begin{CD} E @> @FVV{\pi}V @> M \end{CD}}
\qqquad
{\folarrowsetup{thickness=0.8pt}
\begin{tikzcd} E \arrow[d, foliated, "\pi"] @> M \end{tikzcd}}
\qqquad
\begin{CD} E @> @FVV{\pi}V @> M \end{CD}
\]
$E \folrightarrow M$,
{\folarrowsetup{shift=0.5}$E \folrightarrow M$,
$E \folrightarrow M$
\end{document}
```

Result:



`\folarrowsetup{gap=0.45}` in the preamble affects the whole document. `\folarrowsetup` used inside a group (`{` and `}`) affects only that group, and the setting of the whole document is restored after the group. The diagrams are, from left to right: the setting of the whole document (`gap=0.45`); in addition `span=1.5`, `lean=0.5`; `thickness=0.8pt` (`tikz-cd`); back to the setting of the whole document. The inline arrows are, from left to right: the setting of the whole document; in addition `shift=0.5` (the strokes are moved toward the tail of the arrow); back to the setting of the whole document. Since the shape keys are not at their defaults, the inline arrows of this example are not typeset with the arrow fonts of section 6.1 but drawn with the strokes over the arrow.

How the width of the arrowhead, which is the reference for the size of the strokes, and the line thickness are determined is described in section 8. In `tikz-cd`, if `thickness` is empty, the strokes are drawn as thick as the shaft of the arrow.

Of the two directions “ $\pm \text{lean} \times \text{direction of the arrow} + \text{normal}$ ”, the strokes take the one that rises to the right. On horizontal arrows they are upright “`///`”, and on vertical arrows they are three nearly horizontal strokes rising to the right.

8 Width of the arrowhead and line thickness

The length and spacing of the strokes are based on the width of the arrowhead, and the line thickness on the thickness of the shaft. These differ greatly among fonts (even at 10pt, the arrowhead width ranges from 2.4pt to 5.6pt) and often cannot be obtained correctly from the font metrics (TFM or `\fontdimen`). So they are determined for the arrow font actually used, in the following order.

1. **Lua $\LaTeX$ with an OpenType math font:** the bounding box of the glyph in the arrow's box is read directly from the font and gives the arrowhead width. The line thickness is the height of the minus sign (U+2212) of the same font. Script glyphs (OpenType ssty) used at script sizes and the large glyphs of `\Big` are also read (`folarrow.lua`).
2. **Xe $\LaTeX$ with an OpenType math font:** the bounding box of the glyph is read with `\XeTeXglyphbounds`. Script glyphs and the large glyphs of `\Big` cannot be read, so the values of the base glyph are used.
3. **Type 1 math fonts** (pdf $\LaTeX$, up $\LaTeX$, Lua $\LaTeX$): the measured data (`folarrow-fonts.def`) is looked up. The font that has the arrow characters is found from the math code of `\rightarrow` and the delimiter code of `\downarrow` (some packages, such as `drm` and `euler`, take the arrows from fonts other than the symbol font). The data has, for each font name (the base part of `\fontname`), the ratios of the arrowhead width and the shaft thickness to the quad. Fonts whose shapes differ with optical size, such as `cmsy7` and `cmsy10`, have entries under their own names.
4. **Otherwise:** the values are estimated from the height of the box of `\rightarrow`, the height of the math axis and the rule thickness of the math extension font. The ratios to the quad are limited to avoid extreme values.

For OpenType fonts whose minus sign is not as thick as the shaft of the arrow (Asana Math, GFS Neohellenic Math, the `\Big` glyphs of Minion Math and of KpMath), a correction measured from the fonts is applied.

8.1 Position of the strokes

The strokes are placed at the center of the arrow's ink (on the shaft, at the middle of the arrow's length). The center of the arrow's box and the center of its ink often differ (for the base of `\foldownarrow` and `\foluparrow` and for `\longrightarrow`, by about 0.1–0.7pt toward the head, and with some fonts by 0.1–0.3pt across the shaft). A left arrow is not necessarily a mirrored right arrow, so its offsets differ from those of the right arrow. So, for each of eight kinds of arrows (`\longrightarrow`, `\longleftarrow`, the right and left stretchable arrows, the down and up arrows of `\Big`, and the bases of `\foldownarrow` and `\foluparrow`), the measured data has the difference between the center of the box and the center of the ink in the two directions along and across the arrow, and the strokes are moved by that amount. With Lua $\LaTeX$ and Xe $\LaTeX$, the height of the strokes on horizontal arrows is the vertical center of the arrowhead taken from the glyph's bounding box.

The base of `\foldownarrow` and `\foluparrow` is a delimiter 1.25 times the quad of the symbol font high. The width of the empty delimiter added by `\right`. (`\nulldelimiterspace`) is set to 0, and `\delimiterfactor` is set to 1000 and `\delimitershortfall` to 0, so that the choice of the delimiter glyph does not depend on the font size or the text font.

On stretchable arrows (horizontal arrows of `amscd`, `\xrightarrow`), the distribution of the `\cleaders` that fill the shaft changes with the length of the arrow, so with some fonts an offset of about 0.1pt remains (0.14pt with STIX, 0.17pt with bold `pxfonts`).

In the arrow fonts of section 6.1, the strokes are placed at the center of the ink computed from the outline of the glyph. Measuring the arrows in four directions of 10pt Computer Modern, `newtxmath`, `newpxmath` and `txfonts` in 2400dpi images, the offset from the center was at most 0.015pt (half a pixel) both along and across the arrow. For the OTF fonts for OpenType math fonts, the 48 fonts other than XITS Math Bold were typeset with Lua $\LaTeX$ and Xe $\LaTeX$ and the arrows in four directions measured in the same way. The offset across the arrow was at most 0.03pt for all fonts, and along the arrow at most 0.03pt except for the nine weights of Pennstander Math and Luciole Math Bold. These ten fonts have long barbs reaching back from the arrowhead, which enter the range where the measurement looks for the strokes (25–75% of the arrow's length), so they cannot be measured from images. So, for the glyphs of all fonts, the original arrow was subtracted from the glyph to extract the strokes, and their center

was compared in outline coordinates: the offset was at most 0.0002em (the rounding of coordinates to integers).

In the heavy weights of Pennstander Math (Bold and heavier), the shaft is about as thick as the spacing of the strokes, so the strokes appear joined (this follows from the rules of the shape). With Lua $\text{\LaTeX}$, the extensible $\uparrow$ of Luciole Math gives the error “Extensible recipe has no fixed parts” (a problem of the font; it happens without `\folarrow` too). The glyphs of the OTF fonts do not use this delimiter, so they are not affected, but a drawn `\foluparrow` (with `glyphs=false` etc., including the light version `lite`, which always draws) gives this error.

8.2 Ends of the strokes

The ends of the strokes follow the shape of the arrow’s tail (the end of the shaft): round ends for arrows with a round tail, butt ends otherwise. A round end extends half the line thickness beyond the end, so round strokes are drawn shorter by that amount to keep the same visual length.

- `amscd`, inline math: the measured data has the shape of the tail for each way the arrow is actually built (`\longrightarrow`, stretchable arrows, `\Big\downarrow`, the base of `\foldownarrow`). For example, in Computer Modern the horizontal arrows have a round tail (the end of `\relbar` is round) and `\Big\downarrow` has a flat top, so the strokes have round ends on horizontal arrows and butt ends on vertical arrows. OpenType math fonts are looked up in a table by font name (round for Latin Modern Math and New Computer Modern Math, butt for the others). Fonts not in the table get butt ends.
- `xy-pic`: the lines of `xy-pic` have round ends, so the strokes have round ends.
- `tikz-cd`: the line cap setting of the arrow (`line cap`) is used as it is. The default of `tikz-cd` is butt.

8.3 Verification

For all fonts (68) of “Fonts with Math Support” of The $\text{\LaTeX}$ Font Catalogue (<https://tug.org/FontCatalogue/mathfonts.html>), OpenType math fonts (Latin Modern Math, New Computer Modern Math, STIX Two Math, $\text{\TeX}$ Gyre Termes, Pagella, Bonum, Schola and DejaVu Math, Libertinus Math, Fira Math, XITS Math, Asana Math, Erewhon Math, KpMath, GFS Neohellenic Math, Minion Math), and the bold versions (`\boldmath`) of Computer Modern, Latin Modern, `newtx`, `txfonts`, `newpx` and `pxfonts`, the arrows were actually typeset and rendered at 2400 dpi, and the extent of the ink was measured and compared with the values computed by `\folarrow`. The sizes were 10 pt, 10.95 pt and 12 pt, the styles text, script and scriptscript, and the vertical arrows `\Big\downarrow` and the base of `\foldownarrow`. Type 1 fonts were typeset with pdf $\text{\LaTeX}$ and up $\text{\LaTeX}$, OpenType fonts with Lua $\text{\LaTeX}$ and Xe $\text{\LaTeX}$.

The table below gives the largest difference (%) between the computed and the measured values; values above 3% are in bold. The “Center” column is the largest offset (pt) between the center of the strokes and the center of the arrow’s ink (along and across the arrow, over all sizes of the eight kinds of arrows); values above 0.05 pt are in bold. The “Ends” column is “ok” if the measured shape of the arrow’s tail (round or flat) and the shape of the stroke ends chosen by `\folarrow` agree for all four kinds of arrows (`\longrightarrow`, stretchable arrows, `\Big\downarrow`, the base of `\foldownarrow`). Only the following exceed 3%.

- **PK fonts** (boisik, Computer Concrete, `drm`): bitmap fonts made with METAFONT, whose pixel steps (0.12 pt at 600 dpi) become errors in the dimensions as they are. This is a property of the fonts themselves.
- **Script glyphs** (STIX Two Math and Erewhon Math with Xe $\text{\LaTeX}$): Xe $\text{\TeX}$ cannot read script glyphs, so the values are off by up to 20% at script sizes. With Lua $\text{\LaTeX}$ they are correct.
- **Not installed**: URW Garamond (`mathdesign`) and `garamondx` are non-free fonts that are not in the environment used for the verification ($\text{\TeX}$ Live 2026), so they were not checked.

The table and the data were made from the results of actually typesetting and measuring the arrows.

Font (setting)	Engine	Horizontal		Vertical		Center	Ends	Note
		width	thick.	width	thick.	pt		
<code>antykwaterunska</code>	pdf	1	0	0	0	0.05	ok	

Font (setting)	Engine	Horizontal		Vertical		Center pt	Ends	Note
		width	thick.	width	thick.			
antykwaterunska	up	1	0	0	0	0.06	ok	
antykwaterunskacondensed	pdf	1	0	0	0	0.02	ok	
antykwaterunskacondensed	up	1	0	0	0	0.02	ok	
antykwaterunskalight	pdf	1	0	0	0	0.07	ok	
antykwaterunskalight	up	1	0	0	0	0.07	ok	
antykwaterunskalightcondensed	pdf	1	0	0	0	0.02	ok	
antykwaterunskalightcondensed	up	1	0	0	0	0.03	ok	
arev	pdf	1	0	0	0	0.03	ok	
arev	up	1	0	0	0	0.05	ok	
Asana-Math.otf	Lua	1	0	1	0	0.03	ok	
Asana-Math.otf	Xe	1	0	1	0	0.03	ok	
asanamath	Lua	1	0	1	0	0.03	ok	
baskervaldx	pdf	0	0	0	0	0.02	ok	
baskervaldx	up	0	0	0	0	0.02	ok	
baskervillef	pdf	0	0	0	0	0.02	ok	
baskervillef	up	0	0	0	0	0.02	ok	
boisik	pdf	12	27	0	0	0.20	ok	PK font
boisik	up	12	27	0	0	0.29	ok	PK font
charterbt	pdf	2	0	0	0	0.02	ok	
charterbt	up	2	0	0	0	0.05	ok	
cochineal	pdf	0	0	0	0	0.02	ok	
computerconcrete	pdf	1	10	0	0	0.05	ok	PK font
computerconcrete	up	1	10	0	0	0.14	ok	PK font
computerconcreteeuler	pdf	0	0	0	0	0.03	ok	
computerconcreteeuler	up	0	0	0	0	0.04	ok	
computermodern	pdf	0	0	0	0	0.03	ok	
computermodern	up	0	0	0	0	0.04	ok	
computermodern-bold	pdf	0	0	0	0	0.03	ok	
computermodern-bold	up	0	0	0	0	0.03	ok	
computermodernbright	pdf	0	0	0	0	0.02	ok	
computermodernbright	up	0	0	0	0	0.03	ok	
domitian	pdf	1	0	0	0	0.02	ok	
domitian	up	1	0	0	0	0.04	ok	
drm	pdf	3	13	0	6	0.47	ok	PK font
drm	up	0	5	0	6	0.47	ok	PK font
ebgaramond	pdf	0	0	0	0	0.02	ok	
ebgaramond	up	0	0	0	0	0.02	ok	
erewhon	pdf	0	0	0	0	0.02	ok	
erewhon	up	0	0	0	0	0.02	ok	
erewhon-math	Lua	2	0	1	0	0.03	ok	
Erewhon-Math.otf	Lua	2	0	1	0	0.03	ok	
Erewhon-Math.otf	Xe	6	18	1	0	0.03	ok	script glyphs
eulervm	pdf	0	0	0	0	0.02	ok	
europeancomputerconcrete	pdf	1	10	0	0	0.05	ok	PK font
europeancomputerconcrete	up	1	10	0	0	0.14	ok	PK font
europeancomputerconcreteeuler	pdf	0	0	0	0	0.03	ok	
europeancomputerconcreteeuler	up	0	0	0	0	0.04	ok	
europeancomputermodern	pdf	0	0	0	0	0.03	ok	
europeancomputermodern	up	0	0	0	0	0.04	ok	
europeancomputermodernbright	pdf	0	0	0	0	0.02	ok	
europeancomputermodernbright	up	0	0	0	0	0.03	ok	
Fira Math	Lua	2	0	1	0	0.02	ok	
FiraMath-Regular.otf	Lua	2	0	1	0	0.02	ok	
FiraMath-Regular.otf	Xe	2	0	1	0	0.02	ok	
firasansmath	Lua	2	0	1	0	0.02	ok	
firasansnewtxsf	pdf	0	0	0	0	0.02	ok	
firasansnewtxsf	up	0	0	0	0	0.02	ok	
fourier-only	pdf	1	0	0	0	0.04	ok	
garamond	pdf	—	—	—	—	—	—	not installed
garamond	up	—	—	—	—	—	—	not installed
garamondexpertmathdesign	pdf	—	—	—	—	—	—	not installed
garamondexpertmathdesign	up	—	—	—	—	—	—	not installed

Font (setting)	Engine	Horizontal		Vertical		Center pt	Ends	Note
		width	thick.	width	thick.			
garamondexpertnewtxmath	pdf	—	—	—	—	—	—	not installed
garamondexpertnewtxmath	up	—	—	—	—	—	—	not installed
gfsartemis	pdf	1	0	0	0	0.03	ok	
gfsartemis	up	1	0	0	0	0.02	ok	
gfsartemisaeuler	pdf	0	0	0	0	0.03	ok	
gfsartemisaeuler	up	0	0	0	0	0.04	ok	
gfsneohellenic	pdf	0	0	0	0	0.03	ok	
gfsneohellenic	up	0	0	0	0	0.04	ok	
gfsneohellenicmath	Lua	1	2	1	0	0.04	ok	
GFSNeohellenicMath.otf	Lua	1	2	1	0	0.03	ok	
GFSNeohellenicMath.otf	Xe	1	2	1	0	0.03	ok	
heuristica	pdf	0	0	0	0	0.02	ok	
heuristica	up	0	0	0	0	0.02	ok	
iwona	pdf	1	0	0	0	0.03	ok	
iwona	up	1	0	0	0	0.03	ok	
iwonacondensed	pdf	1	0	0	0	0.03	ok	
iwonacondensed	up	1	0	0	0	0.02	ok	
iwonalight	pdf	1	0	0	0	0.02	ok	
iwonalight	up	1	0	0	0	0.02	ok	
iwonalightcondensed	pdf	1	0	0	0	0.02	ok	
iwonalightcondensed	up	1	0	0	0	0.02	ok	
kerkis	pdf	1	0	0	0	0.03	ok	
kerkis	up	1	0	0	0	0.02	ok	
KpMath-Regular.otf	Lua	2	0	1	1	0.02	ok	
KpMath-Regular.otf	Xe	2	0	0	1	0.02	ok	
kpsansserif	pdf	0	0	0	0	0.04	ok	
kpsansserif	up	0	0	0	0	0.05	ok	
kpserif	pdf	0	0	0	0	0.04	ok	
kpserif	up	0	0	0	0	0.05	ok	
kurier	pdf	1	0	0	0	0.03	ok	
kurier	up	1	0	0	0	0.03	ok	
kuriercondensed	pdf	1	0	0	0	0.03	ok	
kuriercondensed	up	1	0	0	0	0.02	ok	
kurierlight	pdf	1	0	0	0	0.02	ok	
kurierlight	up	1	0	0	0	0.02	ok	
kurierlightcondensed	pdf	1	0	0	0	0.02	ok	
kurierlightcondensed	up	1	0	0	0	0.02	ok	
Latin Modern Math	Lua	1	0	1	0	0.02	ok	
latinmodern-math.otf	Lua	1	0	1	0	0.02	ok	
latinmodern-math.otf	Lua	1	0	1	0	0.02	ok	
latinmodern-math.otf	Xe	1	0	0	0	0.02	ok	
latinmodernroman	pdf	0	0	0	0	0.03	ok	
latinmodernroman	up	0	0	0	0	0.02	ok	
latinmodernroman-bold	pdf	0	0	0	0	0.02	ok	
latinmodernroman-bold	up	0	0	0	0	0.03	ok	
Libertinus Math	Lua	1	0	1	0	0.01	ok	
LibertinusMath-Regular.otf	Lua	1	0	1	0	0.01	ok	
LibertinusMath-Regular.otf	Xe	1	0	1	0	0.02	ok	
libertinusserif	pdf	0	0	0	0	0.03	ok	
libertinusserif	up	0	0	0	0	0.04	ok	
linuxlibertine	pdf	0	0	0	0	0.02	ok	
linuxlibertine	up	0	0	0	0	0.02	ok	
lxfonts	pdf	0	0	0	0	0.02	ok	
lxfonts	up	0	0	0	0	0.03	ok	
Minion Math	Lua	1	0	1	0	0.02	ok	
Minion Math	Lua	1	0	1	0	0.02	ok	
Minion Math	Xe	1	0	1	0	0.02	ok	
mlmodern	pdf	0	0	0	0	0.03	ok	
mlmodern	up	0	0	0	0	0.04	ok	
NewCMMath-Regular.otf	Lua	1	0	1	0	0.02	ok	
NewCMMath-Regular.otf	Xe	1	0	0	0	0.02	ok	
newcomputermodernroman	Lua	1	0	1	0	0.02	ok	

Font (setting)	Engine	Horizontal		Vertical		Center pt	Ends	Note
		width	thick.	width	thick.			
newpx	pdf	1	0	0	0	0.03	ok	
newpx	up	1	0	0	0	0.03	ok	
newpx-bold	pdf	1	0	0	0	0.07	ok	
newpx-bold	up	1	0	0	0	0.07	ok	
newpxeuler	pdf	1	0	0	0	0.03	ok	
newpxeuler	up	1	0	0	0	0.03	ok	
newtx	pdf	0	0	0	0	0.02	ok	
newtx	up	0	0	0	0	0.02	ok	
newtx-bold	pdf	0	0	0	0	0.04	ok	
newtx-bold	up	0	0	0	0	0.06	ok	
newtxsf	pdf	0	0	0	0	0.02	ok	
notoserif	pdf	0	0	0	0	0.02	ok	
notoserif	up	0	0	0	0	0.02	ok	
pxfonts	pdf	0	0	0	0	0.02	ok	
pxfonts	up	0	0	0	0	0.02	ok	
pxfonts-bold	pdf	1	0	0	0	0.17	ok	
pxfonts-bold	up	1	0	0	0	0.09	ok	
sansmathfonts	pdf	0	0	0	0	0.02	ok	
sansmathfonts	up	0	0	0	0	0.02	ok	
scholax	pdf	1	0	1	0	0.04	ok	
scholax	up	1	0	1	0	0.03	ok	
step	pdf	1	0	0	0	0.14	ok	
step	up	1	0	0	0	0.14	ok	
stickstoo	pdf	1	0	0	0	0.02	ok	
stickstoo	up	1	0	0	0	0.02	ok	
stix	pdf	1	0	0	0	0.14	ok	
stix	up	1	0	0	0	0.14	ok	
STIX Two Math	Lua	1	2	1	1	0.02	ok	
stix2	pdf	1	0	1	1	0.02	ok	
stix2	up	1	0	1	1	0.02	ok	
STIXTwoMath-Regular.otf	Lua	1	2	1	1	0.02	ok	
STIXTwoMath-Regular.otf	Xe	11	20	1	1	0.02	ok	script glyphs
TeX Gyre Bonum Math	Lua	2	0	1	0	0.02	ok	
TeX Gyre DejaVu Math	Lua	0	0	0	0	0.06	ok	
TeX Gyre Pagella Math	Lua	1	0	1	0	0.06	ok	
TeX Gyre Schola Math	Lua	2	0	1	0	0.02	ok	
TeX Gyre Termes Math	Lua	1	0	0	0	0.03	ok	
texgyrebonum-math.otf	Lua	2	0	1	0	0.02	ok	
texgyrebonum-math.otf	Xe	1	0	1	0	0.03	ok	
texgyredejavu-math.otf	Lua	0	0	0	0	0.06	ok	
texgyredejavu-math.otf	Xe	0	0	0	0	0.05	ok	
texgyrepagella-math.otf	Lua	1	0	1	0	0.06	ok	
texgyrepagella-math.otf	Xe	1	0	1	0	0.07	ok	
texgyreschola-math.otf	Lua	2	0	1	0	0.02	ok	
texgyreschola-math.otf	Xe	2	0	1	0	0.02	ok	
texgyretermes-math.otf	Lua	1	0	0	0	0.03	ok	
texgyretermes-math.otf	Xe	1	0	0	0	0.02	ok	
txfonts	pdf	1	0	0	0	0.03	ok	
txfonts	up	1	0	0	0	0.02	ok	
txfonts-bold	pdf	1	0	0	0	0.04	ok	
txfonts-bold	up	1	0	0	0	0.04	ok	
urwnimbusroman	pdf	0	0	0	0	0.02	ok	
urwnimbusroman	up	0	0	0	0	0.03	ok	
urwpalladio	pdf	1	0	0	0	0.02	ok	
urwpalladio	up	1	0	0	0	0.02	ok	
urwschoolbookl	pdf	1	0	0	0	0.04	ok	
urwschoolbookl	up	1	0	0	0	0.06	ok	
utopia-fouriermath	pdf	1	0	0	0	0.04	ok	
utopia-fouriermath	up	1	0	0	0	0.06	ok	
utopia-mathdesign	pdf	2	0	0	0	0.02	ok	
utopia-mathdesign	up	2	0	0	0	0.05	ok	
xits	Lua	1	0	1	0	0.02	ok	

Font (setting)	Engine	Horizontal		Vertical		Center pt	Ends	Note
		width	thick.	width	thick.			
XITS Math	Lua	1	0	1	0	0.02	ok	
XITSMath-Regular.otf	Lua	1	0	1	0	0.02	ok	
XITSMath-Regular.otf	Xe	1	0	1	1	0.02	ok	

9 Requirements

Tested with pdfL^AT_EX, LuaL^AT_EX, upL^AT_EX+dvipdfmx and XeL^AT_EX (T_EX Live 2025 and 2026). L^AT_EX 2022-06-01 or later is required (for `\fpeval` and `\DeclareKeys`). In T_EX Live 2025 and 2026, however, xy-pic itself cannot be used with LuaL^AT_EX because of an xypdf error (unrelated to `folarrow`).

10 Notes

- **Key values:** a value that is not a number or is out of range (see the table of section 7) gives an error showing the name and value of the key, and the previous value is kept.
- **Headings and the table of contents:** the commands can be used in the arguments of `\section`, `\caption`, etc. (they are robust). In `hyperref` bookmarks, the inline arrows become the characters $\rightarrow\leftarrow\uparrow$, and the stretchable arrows become $\rightarrow$ and $\leftarrow$ without the labels.
- **Loading order:** `amscd`, `xy-pic`, `tikz-cd` and `unicode-math` may be loaded before or after `folarrow`.
- **dvips:** a package cannot make dvips read a map file, so enable `folarrow.map` with the first command below (or, temporarily, use the second).

```
updmap-user --enable Map=folarrow.map
dvips -u +folarrow.map file.dvi
```

Otherwise the inline arrows are replaced by other characters ($\Gamma\Delta\Theta\Lambda$ with Computer Modern). The package issues a warning about this in documents that use dvips. If you do not use the arrow fonts, specify `glyphs=false`.

- **Japanese typesetting:** the package works with pL^AT_EX, upL^AT_EX, LuaT_EX-ja (including with `unicode-math`) and XeL^AT_EX with `xeCJK`. In vertical writing with pT_EX/upT_EX and dvipdfmx, the drawn strokes are rotated for vertical writing. To typeset diagrams such as `tikz-cd` in vertical writing, put them in a horizontal box such as `\hbox{\yoko ...}` (`tikz-cd` in vertical writing breaks T_ik_Z itself).
- **babel:** with `spanish`, `<` and `>` become active characters and `amscd` itself cannot be used (this happens without `folarrow` too). With the `es-noquoting` option or `\deactivatequoting`, `@F>>>` etc. can be used. To use `tikz-cd` with `german`, `russian`, etc., `\usetikzlibrary{babel}` is needed (a limitation of `tikz-cd`).
- **Color:** for arrows inside `\color` and colored arrows of `tikz-cd`, the strokes have the same color. `\ar@{red}` of `xy-pic` does not pass its color to `|@{fol}` (the same as `|@{>}` etc.; a property of `xy-pic`), so write `\ar@{red}[r]|*{red}\dir{fol}`.
- **Line styles of tikz-cd:** on dashed and dotted arrows, the strokes are solid.
- **Large sizes:** the drawn vertical arrows and the vertical arrows of `amscd` are large delimiters, so at large sizes they stay small without `exscale` (loaded by `amsmath`). Load `amsmath` or `exscale`. The horizontal arrows of `amscd` are never shorter than `\longrightarrow` (at large sizes the 2.5pc of `\minCDarrowwidth` is too short, and the strokes would overlap the arrowhead).
- **Short arrows:** if arrows are made very short, for example by decreasing `\minCDarrowwidth`, with `column sep=tiny` of `tikz-cd` or `@C=0.5pc` of `xy-pic`, the strokes look as if they stick out of the arrow.
- **Tagged PDF:** with `\DocumentMetadata{tagging=on}` etc., the drawn strokes are treated as artifacts, not figures.

- **Processing time:** when the inline arrows are typeset with the glyphs of the fonts (the default), the time is almost the same as without `folarrow`. When they are drawn (`glyphs=false`, `lite`, math fonts not in the tables), each arrow takes a dozen or so milliseconds.

11 License and author

`folarrow.sty`, `folarrow.lua`, the `.def` files, the documentation and the tools are distributed under the MIT License (`LICENSE`). The arrow fonts (`fonts/`) were made from other fonts and keep the licenses of their original fonts (SIL Open Font License 1.1, GUST Font License, GNU GPL with font exception). The original font, original copyright notice and license of each font are listed in `licenses/FONTS.md`, and the license texts are in `licenses/`.

Author: Yuwsuke Kieda